

**IN THE CLAIMS**

Please amend the claims as follows.

For the Examiner's convenience, a list of all claims is included below.

1. (Original) A method for creating a state machine for probability estimation, the method comprising:

assigning probabilities to states of a look up table (LUT), including setting a probability for each state  $i$  of the states equal to the highest probability of the LPS multiplied by the adaptation rate to the power  $i$ , where  $i$  is a number for a given state and the adaptation rate is smaller than 1;

generating state transitions for states in the LUT to be transitioned to upon observing an MPS and an LPS, wherein the next state to which the state machine transitions from a current state when an MPS is observed is a next state higher than the current state if the current state is not the highest state and is the current state if the current state is the highest state, and further wherein the next state to which the state machine transitions from a current state when an LPS is observed for a plurality of states is a rounded version of a result of calculating:

number of current state+log((probability of the current state\*the adaptation rate+(1-the adaptation rate))/probability of the current state)/log(the adaptation rate).

2. (Original) The method defined in Claim 1 wherein the rounded version of the result is such that on average any rounding introduced when generating next states for when an LPS is observed is substantially zero.

3. (Original) The method defined in Claim 1 wherein the LUT has a plurality of values associated with each state, wherein each of the plurality of values approximates a product of an expected interval range multiplied by a probability associated with the state.

4. (Original) The method defined in Claim 3 wherein a value associated with a state is obtained by the product of  $N/(2*M*\log((j+M+1)/(j+M)))$  with a probability associated with the state, rounded to an integer number, where j represents a column index in the array, M is the number of columns in the array, and N is a constant.

5. (Original) The method defined in Claim 3 wherein at least one of the plurality of values for at least one of the plurality of states is clipped to a predetermined number.

6. (Original) The method defined in Claim 5 wherein the predetermined number enables having at most one iteration of renormalization during coding of an MPS.

7. (Original) The method defined in Claim 4 wherein  
the number of states in the LUT is 63,  
the adaptation rate is equal to  $0.5/0.01875$  to the power  $1.0/63$ , and  
the highest probability of an LPS is 0.5, and  
the number of columns in the array is 4, and  
the values in a first column in the array are clipped to  $N/4$ .

8. (Original) The method defined in Claim 7 wherein the number N is 512.

9. (Original) An arithmetic encoder comprising:

a probability estimator to generate a probability estimate that each event of an event sequence has a particular value, wherein the probability estimator generates the probability estimate in response to corresponding context information for said each event using a probability estimation state machine created by assigning probabilities to states of a look up table (LUT), including setting a probability for each state  $i$  of the states equal to the highest probability of the LPS multiplied by the adaptation rate to the power  $i$ , where  $i$  is a number for a given state and the adaptation rate is smaller than 1;

generating state transitions for states in the LUT to be transitioned to upon observing an MPS and an LPS, wherein the next state to which the state machine transitions from a current state when an MPS is observed is a next state higher than the current state if the current state is not the highest state and is the current state if the current state is the highest state, and further wherein the next state to which the state machine transitions from a current state when an LPS is observed for a plurality of states is a rounded version of a result of calculating:  
number of current state+log(probability of the current state\*the adaptation rate+(1-the adaptation rate))/probability of the current state)/log(the adaptation rate); and

a coding engine coupled to the probability estimator to generate zero or more bits of an information sequence in response to each event and its corresponding probability estimate.

10. (Original) The arithmetic encoder defined in Claim 9 wherein the rounded version of the result is such that on average any rounding introduced when generating next states for when an LPS is observed is substantially zero.

11. (Currently Amended) The arithmetic encoder defined in Claim 9 wherein the LUT has a plurality of values associated with each state arithmetic encoder, wherein each of the plurality of values approximates a product of an expected interval range multiplied by a probability associated with the state.

12. (Original) The arithmetic encoder defined in Claim 11 wherein a value associated with a state is obtained by the product of  $N/(2*M*\log((j+M+1)/(j+M)))$  with the probability associated with the state, rounded to an integer number, where j represents a column index in the array, M is the number of columns in the array, and N is a constant.

13. (Original) The arithmetic encoder defined in Claim 11 wherein at least one of the plurality of values for at least one of the plurality of states is clipped to a number.

14. (Original) The arithmetic encoder defined in Claim 13 wherein the number enables having at most one iteration of renormalization during the coding of an MPS.

15. (Original) The arithmetic encoder defined in Claim 9 wherein the number of states in the LUT is 63, the adaptation rate is equal to  $0.5/0.01875$  to the power  $1.0/63$ , and the highest probability of an LPS is 0.5, and the number of columns in the array is 4, and the values in a first column in the array are clipped to  $N/4$ .

16. (Original) The arithmetic encoder defined in Claim 15 wherein the number N is 512.

17. (Currently Amended) An arithmetic decoder comprising:  
a probability estimator to generate a probability estimate that an event of an event sequence has a particular value, wherein the probability estimator generates the probability estimate in response to corresponding context information for said event of the event sequence using a probability estimation state machine created by

assigning probabilities to states of the probability estimation state machine implemented with a look up table (LUT), including setting a probability for each state i of the states equal to the highest probability of the LPS multiplied by the adaptation rate to the power i, where i is a number for a given state and the adaptation rate is smaller than 1, and

generating state transitions for states of the probability estimation state machine in the LUT to be transitioned to upon observing an MPS and an LPS, wherein the next state to which the state machine transitions from a current state when an MPS is observed is a next state higher than the current state if the current state is not the highest state and is the current state if the current state is the highest state, and further wherein the next state to which the state machine transitions from a current state when an LPS is observed for a plurality of states is a rounded version of a result of calculating:

number of current state+log((probability of the current state\*the adaptation rate+(1-the adaptation rate))/probability of the current state)/log(the adaptation rate); and

a decoding engine coupled to the probability estimator to generate an event of an event sequence in response to its corresponding probability estimate and an information sequence.

18. (Original) The arithmetic decoder defined in Claim 17 wherein the rounded version of the result is such that on average any rounding introduced when generating next states for when an LPS is observed is substantially zero.

19. (Currently Amended) The arithmetic decoder defined in Claim 17 wherein the LUT has a plurality of values associated with each state arithmetic encoder, wherein each of the plurality of values approximates a product of an expected interval range multiplied by a probability associated with the state.

20. (Original) The arithmetic decoder defined in Claim 19 wherein a value associated with a state is obtained by the product of  $N/(2*M*\log((j+M+1)/(j+M))$ ) with the probability associated with the state, rounded to an integer number, where j represents a column index in the array, M is the number of columns in the array, and N is a constant.

21. (Original) The arithmetic decoder defined in Claim 19 wherein at least one of the plurality of values for at least one of the plurality of states is clipped to a number.

22. (Original) The arithmetic decoder defined in Claim 21 wherein the number enables having at most one iteration of renormalization during the coding of an MPS.

23. (Original) The arithmetic decoder defined in Claim 17 wherein the number of states in the LUT is 63,  
the adaptation rate is equal to  $0.5/0.01875$  to the power  $1.0/63$ ,  
the highest probability of an LPS is 0.5,

the number of columns in the array is 4, and  
the values in a first column in the array are clipped to N/4.

24. (Original) The arithmetic decoder defined in Claim 23 wherein the number N is 512.

25. (Currently Amended) A decoding method comprising:  
generating a probability estimate that an event of an event sequence has a particular value, the probability estimate being generated in response to corresponding context information for said each event of the event sequence using a probability estimation state machine created by

assigning probabilities to states of the probability estimation state machine implemented with a look up table (LUT), including setting a probability for each state i of the states equal to the highest probability of the LPS multiplied by the adaptation rate to the power i, where i is a number for a given state and the adaptation rate is smaller than 1, and

generating state transitions for states of the probability estimation state machine in the LUT to be transitioned to upon observing an MPS and an LPS, wherein the next state to which the state machine transitions from a current state when an MPS is observed is a next state higher than the current state if the current state is not the highest state and is the current state if the current state is the highest state, and further wherein the next state to which the state machine transitions from a current state when an LPS is observed for a plurality of states is a rounded version of a result of calculating:

number of current state+log((probability of the current state\*the adaptation rate+(1-the adaptation rate) )/probability of the current state)/log(the adaptation rate); and

generating an event of an event sequence in response to its corresponding probability estimate and an information sequence.